ULDB Flight Software Traceability and Verification Matrix

R & FS #	Section	Description	DTR #	Verification Method	Auditor v
FSW-GEN-00110	3.1.1	There shall be two fully redundant flight computers, each containing an Intel 486 processor, and utilizing a PC-104 buss.	3.4.3.1	inspection	
FSW-GEN-00120	3.1.1	There shall be at least two four port UART cards, a 1553B bus interface card, and a customized TDRSS interface card for serial streaming I and Q data to the TDRSS transceiver.	design	inspection	
FSW-GEN-00130	3.1.1	Each flight computer shall utilize a minimum of 1 hard drive with the capability for daisy chained 2 hard drives for data, software operating system storage.	3.4.2.1	inspection	
FSW-GEN-00210	3.1.2	There shall be two redundant flight computers capable of simultaneous data logging.	design	demonstration	
FSW-GEN-00215	3.1.2	Both flight computers shall utilize the 1553B bus. One designated flight computer shall serve as the Bus Controller (BC). The remaining flight computer shall operate in the Remote Terminal (RT) mode.	implied	demonstration	
FSW-GEN-00220	3.1.2	Both science processors shall be configured as RTs via the redundant 1553B bus.	design	demonstration	
FSW-GEN-00225	3.1.2	The RT flight computer shall receive a health status from the BC. The RT flight computer shall reconfigure itself to the BC in the event of a failure by the acting BC flight computer.	design	demonstration	
FSW-GEN-00230	3.1.2	There shall be a watchdog timer on each flight computer that will reboot the computer in the event of a software lockup condition. The flight software shall reset the watchdog timer periodically.	design	demonstration	
FSW-GEN-00235	3.1.2	There shall be provision made for 2 boot locations on each flight computer to protect against the possibility of software corruption and boot device failures.	design	demonstration	
FSW-GEN-00240	3.1.2	The software or BIOS shall provide automatic cycling between multiple boot locations.	design	demonstration	
FSW-GEN-00245	3.1.2	The software shall provide dedicated serial communications for pre-flight operations and ground diagnostics.	design	demonstration	
FSW-GEN-00250	3.1.2	The flight computer hard drives shall remain in a powered down/power saving mode when data is not being accessed or written to them to conserve power and reduce heat production.	design	demonstration	
FSW-DAT-00110	3.2.1	Flight software shall time-stamp all housekeeping data at the time the flight computer receives it.	design	demonstration	
FSW-DAT-01110	3.2.1.1	The flight computers shall obtain science housekeeping and science data from the redundant science instrument processors via 1553B bus.	3.4.1.1	demonstration	
FSW-DAT-02110	3.2.1.2	The flight computers shall obtain blooncraft housekeeping data from the System and Power Control Unit (PCU) stacks on the same AART RS-	3.4.1.2 3.5.3.2.1 3.5.3.3	demonstration	

		232 port.			
FSW-DAT-02120	3.2.1.2	The flight computers shall obtain GMT time from the GPS unit via an RS-232 port. It is desired to obtain absolute position and attitude (azimuth) from the GPS unit via the same RS-232 port.	3.4.1.2	demonstration	
FSW-DAT-02130	3.2.1.2	The flight computers shall obtain ascent velocity from the GPS unit for autoballasting calculations.	3.4.1.2	demonstration	
FSW-DAT-02140	3.2.1.2	The flight computers shall obtain housekeeping data from the Ballooncraft Rotator via an RS-232 port. Rotator data shall be made switchable between both flight computers in the event of a flight computer failure.	3.4.1.2	demonstration	
FSW-DAT-2150	3.2.1.2	The flight computers shall be capable of obtaining housekeeping data (TBD) from Balloon Control Subsystems (CAP, Universal Terminate Package, ULDBV) via an AART RS-232 port. Balloon Control Subsystem data shall be made switchable between both flight computers in the event of a flight computer failure.	3.4.1.2	demonstration	
FSW-DAT-03110	3.2.1.3	The flight computers shall receive TDRSS Transceiver status data via <TBD>.	design	demonstration	
FSW-DAT-03120	3.2.1.3	The flight computers shall obtain status information from the TDRSS antenna control unit (ACU) via an RS-232 port.	design	demonstration	
FSW-DAT-03130	3.2.1.3	The flight computers shall obtain INMARSAT terminal status data via an RS-232 port.	design	demonstration	
FSW-DAT-03140	3.2.1.3	The flight computers shall obtain ARGOS PTT transmitter status data via an RS-232 port.	design	demonstration	
FSW-DAT-03150	3.2.1.3	The ballooncraft Line of Site (LOS) transmitters shall be capable of flowing science and housekeeping data. A dedicated RS-232 port shall allow LOS data transmissions while a separate RS-232 port shall be provided for receiving LOS commands.	design	demonstration	
FSW-DAT-00210	3.2.2	All housekeeping and science data to be logged to hard drive shall be in binary format and indexed by a 1ms time stamp received from GPS.	design	demonstration	
FSW-DAT-00220	3.2.2	Flight software shall format science, science housekeeping and ballooncraft housekeeping data in CCSDS packets for downlink transmission to all communication links.	design	demonstration	
FSW-DAT-00230	3.2.2	Each CCSDS data packet shall contain 1ms resolution time stamp of packet creation.	3.4.3.1 3.5.3.1.1	demonstration	
FSW-DAT-00240	3.2.2	Each ARGOS PTT ID shall contain a separate CCSDS header and 1ms resolution time stamp. The remaining bytes shall be filled up with data for each ID.	design	demonstration	
FSW-DAT-00310	3.2.3	Data to be logged to hard drive shall be buffered or cached into RAM until 90% of RAM is filled. The RAM contents shall then be written to hard drive. In the event of a reboot or power failure, RAM contents shall be lost.	design	demonstration	
FSW-DAT-00320	3.2.3	Flight software shall log all commands received to hard drive with a 1 ms time GMT stamp index obtained from GPS.	3.4.3.1	demonstration	

FSW-DAT-00330	3.2.3	Flight software shall log all science data to hard drive with a 1 ms time stamp index obtained from GPS.	3.4.3.1	demonstration	
FSW-DAT-00340	3.2.3	Flight software shall log all science housekeeping data to hard drive with a 1 ms time stamp index obtained from GPS.	3.4.3.1	demonstration	
FSW-DAT-00350	3.2.3	Flight software shall log allallooncraft housekeeping data to hard drive with a 1 ms time stamp index obtained from GPS.	3.4.3.1	demonstration	
FSW-DAT-00410	3.2.4	Flight software on both flight computers shall transfer all science, science housekeeping, and ballooncraft housekeeping data formatted in CCSDS packets to the WFF-93 PCM Encoder's RS232 deck. This data transfer shall support the LOS data downlink transmission.	3.4.3.1 3.3.3.1	demonstration	
FSW-DAT-00420	3.2.4	Flight software shall transfer all science, science housekeeping, and ballooncraft housekeeping data to a custom FIFO board for TDRSS transmission. Data shall be transferred in a nature that supports a continuous TDRSS downlink data rate of 50kbps.	3.3.1.1 3.3.1.2 3.4.3.1	demonstration	
FSW-DAT-00430	3.2.4	Flight software shall transfer science housekeeping, and ballooncraft housekeeping data to the INMARSAT terminal in 15 minute transmission intervals.	3.4.3.1	demonstration	
FSW-DAT-00440	3.2.4	Flight software shall transfer science housekeeping, and ballooncraft housekeeping data to the ARGOS PTT for downlink transmissions.	3.4.3.1	demonstration	
FSW-DAT-00450	3.2.4	Playbacks shall utilize the GMT time stamp for indexing data from the hardrive.	3.5.3.1.1 3.4.3.1	demonstration	
FSW-DAT-00460	3.2.4	The flight computers shall forward GMT from GPS to the science CPUs in 1Hz intervals over the 1553B interface.	3.4.3.1	demonstration	
FSW-DAT-00470	3.2.4	The flight computers shall forward MKS pressure, gauge select and GPS vertical velocity to the CAP via an AART line at 1Hz intervals during ascent and TBD Hz during float.	design	demonstration	
FSW-CMD-00010	3.3	All commands shall be categorized by the flight software as Science or ULDB.	design	demonstration	
FSW-CMD-00020	3.3	A backup command decoder shall be capable of decoding commands from TDRSS, INMARSAT and LOS in the format described under 3.3.1.	3.4.4.2	demonstration	
FSW-CMD-00110	3.3.1	The flight software shall encode and decode an 8-byte binary format for all commands sent and received by the flight computer.	design	demonstration	
FSW-CMD-01110	3.3.1.1	There shall be two frame sync bytes, FA(hex) and F3(hex).	design	demonstration	
FSW-CMD-01120	3.3.1.1	There shall be a minimum of 1 byte for the balloon ID and routing address. The first four bits including the LSB shall be the routing address while the last 4 bits including the MSB shall be the balloon ID.	design	demonstration	
FSW-CMD-01130	3.3.1.1	There shall be 1 byte allocated for the ones complement of the balloon ID and routing address.	design	demonstration	
FSW-CMD-01140	3.3.1.1	There shall be 1 byte allocated for the AART address. The MSB must be high at all times.	design	demonstration	

FSW-CMD-01150	3.3.1.1	There shall be 1 byte allocated for the ones complement of the AART address. The MSB must be low at all times.	design	demonstration	
FSW-CMD-01160	3.3.1.1	There shall be 1 byte allocated for the AART command message also called the AART command select. The MSB must be low at all times.	design	demonstration	
FSW-CMD-01170	3.3.1.1	There shall be 1 byte allocated for the ones complement of the AART command message also called the AART command select. The MSB must be 1 at all times.	design	demonstration	
FSW-CMD-00210	3.3.2	The flight software shall process all received commands through asynchronous AART line(s). All transmitted commands shall communicate via an AART and/or 1553b bus.	design	demonstration	
FSW-CMD-00220	3.3.2	The Universal Terminate Package (UTP) shall have a command capability from the flight computer via an AART line.	implied	demonstration	
FSW-CMD-00230	3.3.2	The CAP shall be commandable from the flight computer via an AART line.	design	demonstration	
FSW-CMD-01210	3.3.2.1	The flight software shall be capable of receiving commands via the Telemetry Data Relay Satellite System (TDRSS) network.	design	demonstration	
FSW-CMD-01220	3.3.2.1	The fight software shall be capable of receiving commands via the INMARSAT network.	design	demonstration	
FSW-CMD-01230	3.3.2.1	The INMARSAT command initialization file shall contain all the necessary commands required to initialize and startup the INAMRSAT terminal.	design	demonstration	
FSW-CMD-01240	3.3.2.1	The flight software shall be capable of receiving commands via the Iridium network (TBD).	design	demonstration	
FSW-CMD-02210	3.3.2.2	The flight software shall be capable of receiving commands via the Line of Site (LOS) receivers.	3.3.3.1 3.3.3.2	demonstration	
FSW-CMD-00310	3.3.3	The flight software shall perform error checking as listed in 3.3.4 prior to all command execution.	3.4.3.1	demonstration	
FSW-CMD-00320	3.3.3	The flight software shall execute commands addressed to the flight computer.	implied	demonstration	
FSW-CMD-00330	3.3.3	The flight software shall follow the echo command format (3.3.7) for successful or unsuccessful command receipt confirmation back to the command originator.	3.4.3.1	demonstration	
FSW-CMD-00340	3.3.3	Mission critical commands as defined in the Project Data Base of the operation control center requirements document shall be given execution priority over all other commands.	design	demonstration	
FSW-CMD-00410	3.3.4	The flight software shall use the ones complement in the command format to identify any errors during command transmission.	design	demonstration	
FSW-CMD-00420	3.3.4	The flight software shall validate all commands and ensure that the commands accepted conform to the command format definition as stated in 3.3.1.	3.4.3.1	demonstration	
FSW-CMD-00510	3.3.5	The flight software shall provide the capability to verify the receipt of real-time commands by ground telemetry.	3.4.3.1	demonstration	
FSW-CMD-00520	3.3.5	The flight software shall echo (3.3.6) to the	3.4.3.1	demonstration	

		command originator the status of each command received as successful or unsuccessful.			
FSW-CMD-00610	3.3.6	The flight software command echo shall contain 1 byte of command counter information. The counter shall count up to 1 bytes worth of commands before resetting to 0. The command counter shall keep track of all commands received by the flight computer.	design	demonstration	
FSW-CMD-00620	3.3.6	The flight software command echo shall contain the same 1 byte command message/select as the last command received.	design	demonstration	
FSW-CMD-00630	3.3.6	The flight software command echo shall contain the same 1 byte address of the last command received. This address is the same address required to route commands through the AART bus for source identification.	design	demonstration	
FSW-CMD-00710	3.3.7	All commands received by the flight computer shall be logged to RAM upon receipt. The flight computers shall wait for a routine write to disk command before issuing the write to the hard drive command.	design	demonstration	
FSW-CMD-00720	3.3.7	All data written to hard drive shall have a 1ms resolution time stamp for each line of data recorded. Playbacks shall index the time stamp for start and stop points.	design	demonstration	
FSW-CMD-00730	3.3.7	Commands addressed to the flight computer and science shall be logged a second time when processed or routed to the science subsystem.	design	demonstration	
FSW-CMD-00740	3.3.7	Commands addressed to the CAP, Universal Terminate Package (UTP), Science System stacks, Backup Command stack, Power Control Unit (PCU) stacks, ULDBV, and Rotator shall be logged once.	design		
FSW-CMD-00810	3.3.8	The entire 8-byte command packet shall be forwarded to the appropriate subsystem once the flight software receives a command packet whose routing ID matches that subsystem.	design	demonstration	
FSW-CMD-00820	3.3.8	Command routing locations shall include the flight computer, CAP, UTP, Science system stacks, Backup Command stack, PCU stacks, ULDBV and Rotator.	3.4.3.1	demonstration	
FSW-CMD-00830	3.3.8	The routing address shall be 4 bits in length in the command format yielding a total of 16 unique locations.	design	demonstration	
FSW-MSC-00110	3.4.1	The flight software shall use an automated ballasting function to maintain the balloncraft within a specified altitude range throughout the duration of the mission.	2.2.1.3.2	demonstration	
FSW-MSC-00120	3.4.1	The autoballasting function shall manage the daily allotment of ballast from a configurable file, and keep a daily tally of how much ballast has been dropped for that day. Ballast shall not be dropped if the daily tally equals or exceeds the daily allotment. The daily ballast tally shall be reset to 0 once the next solar noon occurs.	2.2.1.3.2	analysis	

FSW-MSC-00130	3.4.1	The autoballasting function shall recognizing a day as local solar noon to local solar noon as determined by GMT and GPS position.	2.2.1.3.2	analysis	
FSW-MSC-01110	3.4.1.1	The autoballasting algorithm shall use a maximum of 3 MKS pressure sensors at any one time to derive an average pressure reading and compare this to the specified altitude range. This function shall be user configurable.	2.2.1.3.2	demonstration	
FSW-MSC-01120	3.4.1.1	Autoballasting shall not be activated until the ballooncraft has transcended the activation pressure threshold above which ballasting is disabled. The activation pressure threshold shall be user configurable.	2.2.1.3.2	demonstration	
FSW-MSC-01130	3.4.1.1	The amount of time the autoballasting algorithm has between ballast drops, and the number of seconds the ballast valve is allowed to open in a day shall be user configurable.	2.2.1.3.2	demonstration	
FSW-MSC-02110	3.4.1.2	A status file shall be used to identify the current parameter settings of the autoballster. This file shall be used for resetting the autoballaster in the event of a reboot. The ballast configuration file shall be used in the event the autoballasting status file is not available at the time of reboot.	2.2.1.3.2	demonstration	
FSW-MSC-02120	3.4.1.2	A pressure log on hard drive shall be kept every 10 seconds of the average MKS pressure in mbars. This log shall have a minimum of the time of day, pressure (mbars) and the MKS sensor.	design	demonstration	
FSW-MSC-02130	3.4.1.2	A ballast log shall be kept of all ballast drops and/or changes to the autoballast status. The ballast log shall have a minimum of the following information: time of day, last MKS sensor read, pressure (mbars), active high altitude sensor, active mid altitude sensor, low altitude pressure boundary (mbars), high altitude pressure boundary (mbars), wait time (seconds), daily ballast limit (seconds), initial ballast drop amount, new day reset, total ballast dropped today (seconds), longitude (degrees), day change method (software startup, solar noon occurred, day reset via command), last local solar time, day (same or new), acitvation threshold (mbars).	2.2.1.3.2	demonstration	
FSW-MSC-00210	3.4.1.2	The INMARSAT satellite selection algorithm (ISSA) shall select the highest elevation satellite by using a configurable ASCII file to ingest data.	design	demonstration	
FSW-MSC-00220	3.4.1.2	The ISSA shall use GPS position and altitude of the ballooncraft and the latest satellite ephemeral from a configurable ASCII file to compute the highest visible INMARSAT satellite within a given ocean region.	design	analysis	